INSTITUTO TECNOLOGICO DE COSTARICA

Escuelas de computación y administración de empresas

Curso:

Lenguajes de programación

Prof. Andrei fuentes

Tarea programada 3

Integrantes:

Esteban Cespedes Monestel

Julio Calderón

Esteban Aguilar

Contenido

[Descripción del problema 3](#_Toc371630632)

[Diseño del programa 4](#_Toc371630633)

[Tabla dinámica 4](#_Toc371630634)

[Tabla estática 4](#_Toc371630635)

[Librerías implementadas 5](#_Toc371630636)

[Análisis de resultados 5](#_Toc371630637)

[Manual de usuario 6](#_Toc371630638)

[Instalación 6](#_Toc371630639)

[Ejecutar Python 7](#_Toc371630640)

[Directamente desde la Terminal de Ubuntu: 7](#_Toc371630641)

[Ejecutando archivos con código Python desde la terminal: 7](#_Toc371630642)

[Conclusiones 9](#_Toc371630643)

# Descripción del problema

Según las requisiciones se debe recopilar e identificar tanto el ambiente estático como el ambiente dinámico de un programa escrito en el lenguaje de programación sml el cual tendrá una visualización de datos en forma de tabla para su resolución se harán uso de las herramientas e implementaciones convenientes las cuales serán capaces de alcanzar los objetivos propuestos.

# Diseño del programa

Para la implementación de la solución computacional se decidió utilizar el lenguaje de programación Python 3.1. Posteriormente se analizaron los requisitos tomando las siguientes decisiones para la resolución de los mismos:

## Tabla dinámica

1. Se decidió implementar un algoritmo capaz de identificar aquellas variables cuya asignación fuese una constante para posteriormente almacenarlas en una lista enlazada la cual almacenara el valor y la expresión por aparte esto con el fin de resolver operaciones más complejas que requieran expresiones anteriormente definidas para luego ser almacenadas posteriormente.
2. Posteriormente se realizó un algoritmo para procesar variables que tuvieran condicionales específicamente if los cuales se servirán de la lista anteriormente cargada para gestionar variables o valores ya definidos y también algoritmos para la correcta asignación y gestión del valor correspondiente a la expresión que referencian para su posterior almacenaje.
3. Por último se realizaron los algoritmos para la resolución de expresiones let dentro de la asignación de la variable referenciada y ya con esto se procede a implementar un algoritmo que imprima la tabla dinámica como lo es requerida.

## Tabla estática

1. Para su implementación una vez procesada la tabla dinámica se procede a procesar el tipo de cada expresión almacenada en la lista anteriormente definida para posteriormente imprimir en pantalla la expresión correspondiente detallando inmediatamente su tipo cumpliendo así con los requerimientos solicitados.

# Librerías implementadas

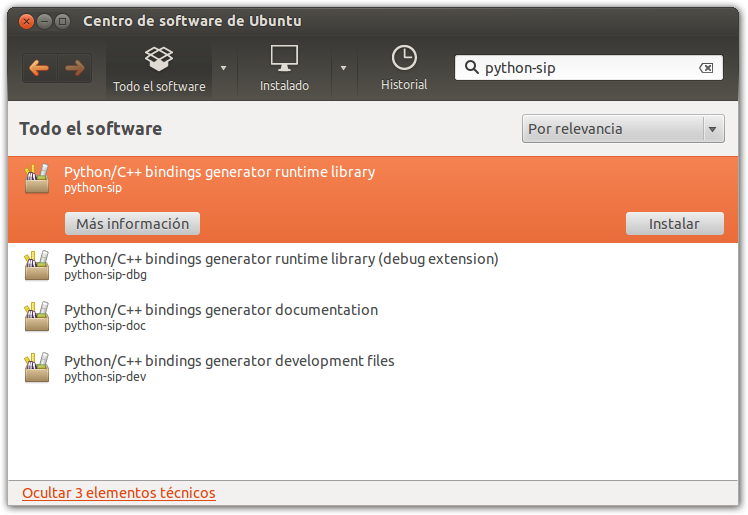
Se importó la librería re la cual está incluida dentro del lenguaje Python 3.1.

# Análisis de resultados

Se logró implementar los dos ambientes requeridos tanto el estático como el dinámico, se tuvieron problemas a la hora de implementar la identificación de tuplas así como la implementación de los procesamientos de los lets dentro del código. También se analizó un bug en el procesamiento de los ifs el cual se logró corregir satisfactoriamente.

# Manual de usuario

## Instalación

1. Descargar el ide del lenguaje de programación Python del siguiente enlace <http://www.python.org/download/other/>
2. O ingresar al software store de Ubuntu digitar Python en la opción de búsqueda e instalar la aplicación

## Ejecutar Python

### Directamente desde la Terminal de Ubuntu:

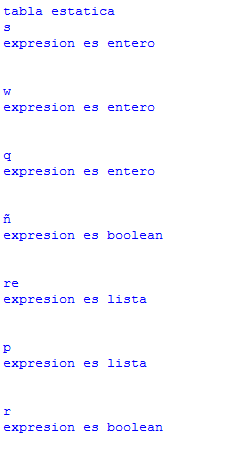
* Para iniciar Python desde la terminal hay que teclear **python** y se pone en marcha el intérprete de Python.
* Para terminar de trabajar con el intérprete hay que teclear **quit()**y regresaremos a la terminal de Ubuntu**.**

### Ejecutando archivos con código Python desde la terminal:

Supongamos que nos encontramos en la terminal de Ubuntu, hemos cambiado de directorio de trabajo y en el actual directorio de trabajo se encuentra un archivo llamado saludo.py y en su interior hay una única línea de texto como la siguiente. print "Hola, mundo". Para ejecutar el archivo estando en la terminal hay que teclear lo siguiente python saludo.py y obtendremos la respuesta automáticamente.

1. Entramos en la terminal y nos desplazamos hasta el directorio de trabajo (Figura 2.1)
2. Generamos el archivo **saludo.py** utilizando el editor **gedit**. Salvamos los cambios y salimos del editor (Figura 2.2).
3. Desde la terminal tecleamos **python saludo.py** y vemos el resultado.





# Conclusiones

La implementación de esta solución computacional reforzó los conocimientos acerca de las técnicas de implementación que realizan los lenguajes de programación para su correcta funcionalidad haciendo esto posible la implementación y resolución de problemas atreves de los mismos.